What is claimed is:

1. A method of executing instructions on a processor, the method comprising:

receiving a first condition code produced by executing a first instruction during a first clock cycle on an array of engines included in the processor;

receiving a second condition code produced by executing a second instruction during a second clock cycle on the array of engines included in the processor; and

executing a logical operator on the first and second condition codes during the second clock cycle on the array of engines included in the processor.

2. The method of claim 1, wherein executing the logical operator includes executing a logical "AND" operation.

3. The method of claim 1, wherein executing the logical operator includes ignoring the first condition code.

4. The method of claim 1, wherein executing the logical operator includes ignoring the second condition code.

5. The method of claim 1, wherein the first instruction is executed on one engine within the array of engines included in the processor.

6. The method of claim 1, wherein the first condition code indicates if the execution of the first instruction produced a numerical zero.

7. The method of claim 1, wherein the first condition code indicates if the execution of the first instruction produced an overflow.

8. The method of claim 1, wherein the first condition code includes data for determining a destination of a data packet.

9. The method of claim 1, wherein a third instruction is executed by the array of engines prior to executing the second instruction.

10. A computer program product, tangibly embodied in an information carrier, for executing instructions on a processor, the computer program product being operable to cause a machine to:

receive a first condition code produced by executing a first instruction during a first clock cycle on an array of engines included in the processor;

receive a second condition code produced by executing a second instruction during a second clock cycle on the array of engines included in the processor; and

execute a logical operator on the first and second condition codes during the second clock cycle on the array of engines included in the processor.

11. The computer program product of claim 10, wherein executing the logical operator includes executing a logical "AND" operation.

12. The computer program product of claim 10, wherein executing the logical operator includes ignoring the first condition code.

13. The computer program product of claim 10, wherein executing the logical operator includes ignoring the second condition code.

14. The computer program product of claim 10, wherein the first instruction is executed on one engine within the array of engines included in the processor.

15. The computer program product of claim 10, wherein the first condition code indicates if the execution of the first instruction produced a numerical zero.

16. The computer program product of claim 10, wherein the first condition code indicates if the execution of the first instruction produced an overflow.

17. The computer program product of claim 10, wherein the first condition code includes data for determining a destination of a data packet.

18. The computer program product of claim 10, wherein a third instruction is executed by the array of engines prior to executing the second instruction.

19. A packet classifier comprises:

a process to receive a first condition code produced by executing a first instruction during a first clock cycle on an array of engines included in a processor;

a process to receive a second condition code produced by executing a second instruction during a second clock cycle on the array of engines included in the processor; and

a process to execute a logical operator on the first and second condition codes during the second clock cycle on the array of engines included in the processor.

20. The packet classifier of claim 19, wherein executing the logical operator includes executing a logical "AND" operation to determine a destination for a packet.

21. The packet classifier of claim 19, wherein executing the logical operator includes ignoring the first condition code.

22. The packet classifier of claim 19, wherein executing the logical operator includes ignoring the second condition code.

23. The packet classifier of claim 19, wherein the first instruction is executed on one engine within the array of engines included in the processor.

24. The packet classifier of claim 19, wherein the first condition code indicates if the execution of the first instruction produced a numerical zero.

25. The packet classifier of claim 19, wherein the first condition code indicates if the execution of the first instruction produced an overflow.

26. The packet classifier of claim 19, wherein the first condition code indicates if the execution of the first instruction produced an overflow.

27. The packet classifier of claim 19, wherein a third instruction is executed by the array of engines prior to executing the second instruction.

28. A system comprising a processor capable of:

receiving a first condition code produced by executing a first instruction during a first clock cycle on an array of engines included in the processor;

receiving a second condition code produced by executing a second instruction during a second clock cycle on the array of engines included in the processor; and

executing a logical operator on the first and second condition codes during the second clock cycle on the array of engines included in the processor.

29. The system of claim 28, wherein executing the logical operator includes executing a logical "AND" operation.

30. The system of claim 28, wherein executing the logical operator includes ignoring the first condition code.

31. A system comprising:

a router including an input port for receiving data packets and an output port for data packet delivering as determined by a switch fabric; and

a processor capable of,

receiving a first condition code produced by executing a first instruction during a first clock cycle on an array of engines included in the processor,

receiving a second condition code produced by executing a second instruction during a second clock cycle on the array of engines included in the processor, and

executing a logical operator on the first and second condition codes during the second clock cycle on the array of engines included in the processor.

32. The system of claim 31, wherein executing the logical operator includes executing a logical "AND" operation.

33. The system of claim 31, wherein executing the logical operator includes ignoring the first condition code.

34. A processor comprising:

an engine having an instruction set including an instruction having a syntax that specifies a logical operation on a condition code determined by executing the instruction and on another condition code determined from a previously executed instruction.

35. The processor of claim 34, wherein the processor includes a second engine.

36. The processor of claim 34, wherein the engine is multithreaded.